

## Re: RDO objects and C#

---

*Source:*

<http://www.tech-archive.net/Archive/Outlook/microsoft.public.outlook.interop/2007-10/msg00030.html>

---

- *From:* AK <[AK@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:AK@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx)>
  - *Date:* Sun, 28 Oct 2007 11:32:00 -0700
- 

Hello Dmitry,

I'm using GC.Collect and modified my code to remove multiple dot notations (e.g. rdoFolder.Items.count) but still I'm getting an error for "MAPI\_E\_CALL\_FAILED".

This is happening especially for RDOMail object.

My code for it reads as

```
=====
Queue DeleteEMailQueue = new Queue();
RDOItems rdoItems = rdoFolder.Items;
for (int i = 0; i < rdoItems.Count; i++)
{
    RDOMail rdoMail = rdoFolder.Items[i];
    try
    {
        // check if email can be deleted
        if (DeleteEmail(rdoMail.UnRead, rdoMail.ReceivedTime,
            rdoMail.Mileage))
        {
            rdoMail.Delete(0);
        }
    }
    catch (Exception ex)
    {
        Util.LogException(ex);
    }
    finally
    {
        System.Runtime.InteropServices.Marshal.ReleaseComObject(rdoMail);
        rdoMail = null;
        GC.Collect();
    }
}
=====
```

The error occurs after reading say around 700/800 emails in one folder.

Re: RDO objects and C#

Can you please suggest any workaround for this?

Thanks and Regards,  
Amit Kathane

"Dmitry Streblechenko" wrote:

You are most likely running out of the RPC channel limit (<http://support.microsoft.com/kb/830829>) – since .Net does not release COM objects immediately, you can easily go over the limit even if you explicitly release COM objects using Marshal.ReleaseCOMObject.

Avoid using multiple dot notation (e.g. rdoFolder.Items.Count) to make sure the compiler does not create implicit variables that you cannot reference and release.

GC.Collect() might help.

Next version of Redemption (send me an e-mail if you want a beta version) will keep track of the open MAPI objects and transparently release oldest unmodified MAPI objects that can be easily reopened on demand (Redemption objects will stay alive of course) when the number of open objects of a particular kind approaches the limit.

Dmitry Streblechenko (MVP)

<http://www.dimastr.com/>

OutlookSpy – Outlook, CDO  
and MAPI Developer Tool

"AK" <AK@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message  
<news:0D7CB539-49DA-4449-A26B-09CB2DB2CD8F@xxxxxxxxxxxxxxxxxxxx>

Hello Dmitry,

Thank you very much for your assistance.

I'm getting following error randomly.

"Error in IMAPIFolder.GetContentsTable: MAPI\_E\_CALL\_FAILED"

Please find below my code

```
=====  
private static void ProcessUserMailbox(string userName)  
{  
    RDOSession rdoSession = new RDOSessionClass();  
    try  
    {  
        // get the admin user name from the configuration file  
        string systemUserName =  
            Properties.Settings.Default.userName;  
        string exchangeSrvrName =
```

## Re: RDO objects and C#

```
Properties.Settings.Default.UserMailboxServer;

if (string.IsNullOrEmpty(systemUserName))
    systemUserName =
    WindowsIdentity.GetCurrent().Name.ToString();

rdoSession.LogonExchangeMailbox(systemUserName,
exchangeSrvrName);
IRDOFolder rootFolder =
rdoSession.GetSharedMailbox(userName).RootFolder;

// iterate through all folders and subfolders
RDOFolders rdoFolders = rootFolder.Folders;
if (rdoFolders != null)
{
    RDOFolder rdoFolder = rdoFolders.GetFirst();
    if (rdoFolder != null)
    {
        // iterate folder list
        for (int i = 0; i < rdoFolders.Count; i++)
        {
            // process the individual folder to search for
            emails that can be deleted
            ProcessFolder(userName, rootFolder.Name + " >
            ",
            rdoFolder);
            rdoFolder = rdoFolders.GetNext();
        }
    }
}
// release the COM object for the root folder.

System.Runtime.InteropServices.Marshal.ReleaseComObject(rootFolder);
rootFolder = null;
}
catch (Exception ex)
{
    StringBuilder strInfo = new
    StringBuilder(string.Format("{0}, Failed to process mailbox for user :
    {0}",
    userName));
    Util.LogMessage(strInfo.ToString());
    Util.LogException(ex);
    strInfo.AppendFormat(ex.Message + Environment.NewLine +
    Util.GetInnerException(ex));
    throw new ApplicationException(strInfo.ToString());
}
finally
{
    rdoSession.Logoff();
}
```

## Re: RDO objects and C#

```
System.Runtime.InteropServices.Marshal.ReleaseComObject(rdoSession);
rdoSession = null;
GC.Collect();
}
}

private static void ProcessFolder(string userName, string
folderPath, RDOFolder rdoFolder)
{
    RDOFolders subFolders;
    RDOFolder subFolder;

    try
    {
        subFolders = rdoFolder.Folders;
        if (subFolders != null)
        {
            subFolder = subFolders.GetFirst();
            if (subFolder != null)
            {
                // iterate sub folder list
                for (int i = 0; i < subFolders.Count; i++)
                {
                    // process the subfolder
                    ProcessFolder(userName, rdoFolder.Name + "
>
", subFolder);
                    subFolder = subFolders.GetNext();
                }
            }
        }
        // if found an individual folder (rdoFolder) without
the
subfolder then release the subfolder and subfolders COM objects

//System.Runtime.InteropServices.Marshal.ReleaseComObject(subFolder);
subFolder = null;

System.Runtime.InteropServices.Marshal.ReleaseComObject(subFolders);
subFolders = null;

// if there are no sub folders then process the
rdoFolder and iterate through all items
// process the individual items/emails in the folder
for (int i = 0; i < rdoFolder.Items.Count; i++)
{
    RDOMail rdoMail = rdoFolder.Items[i];
    try
    {
        // check if email can be deleted
        if (DeleteEmail(rdoMail.UnRead,
```

Re: RDO objects and C#

```
rdoMail.ReceivedTime, rdoMail.Mileage))
{
Util.LogMessage(string.Format("{0}, Email
with subject '{1}' Read : '{2}' Received Time : '{3}' Mileage : '{4}' is
deleted.", userName, rdoMail.Subject, !(rdoMail.UnRead),
rdoMail.ReceivedTime, rdoMail.Mileage));
if (DeleteMail)
rdoMail.Delete(0); //dfSoftDelete (0) –
default. Deletes the item. Can still be recoverable if retention policy is
set on Exchange Server.

//dfMoveToDeletedItems
(1) – the item is moved to the Deleted Items folder
//dfHardDelete (2) –
Exchange only. Permanently deletes the item; will not be recoverable
}
}
catch (Exception ex)
{
// in case of exception for processing
individual mail log the exception and process next email
Util.LogException(ex);
}
finally
{

//System.Runtime.InteropServices.Marshal.ReleaseComObject(rdoMail);
rdoMail = null;
}
}
}
catch (Exception ex)
{
StringBuilder strInfo = new
StringBuilder(string.Format("{1}, Failed to process folder : {0} for user
:
{1}", rdoFolder.Name, userName));
Util.LogMessage(strInfo.ToString());
Util.LogException(ex);
}
finally
{

System.Runtime.InteropServices.Marshal.ReleaseComObject(rdoFolder);
rdoFolder = null;
}
}
=====
```

The above error is coming for some of the emails for user. Can you please let me know what could be the reason of this error.

Re: RDO objects and C#

Thanks and Regards,  
Amit Kathane

"Dmitry Streblechenko" wrote:

1. Exchange transparently manages the location of the mailboxes. It will list all mailboxes in the organization. As an extra check, you might want to read the PR\_EMS\_AB\_HOME\_MDB property using RDOAddressEntry.Fields[] (the property returns the name of the server where the user has a mailbox) – there can be entries that have no mailboxes.
2. Yes, use either (but not both) Logon or LogonExchangeMailbox. Or set the MAPIOBJECT property if you already have a running instance of Outlook (use Namespace.MAPIBJECT) or CDO 1.21 session (Session.MAPIBJECT).

Dmitry Streblechenko (MVP)  
<http://www.dimastr.com/>  
OutlookSpy – Outlook, CDO  
and MAPI Developer Tool

"AK" <AK@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message  
[news:319E9C32-4D06-4142-B3A7-15A4FB7F6581@xxxxxxxxxxxxxxxxxxxx](mailto:news:319E9C32-4D06-4142-B3A7-15A4FB7F6581@xxxxxxxxxxxxxxxxxxxx)

Hello Dmitry,

Thank you very much for your reply.

According to your solution mentioned in one of the above responses i.e.

1. Run your code under the Windows user identity who can access all of the required mailboxes
2. Call RDOSession.Logon if you have an existing profile pointing to that user's mailbox or call RDOSession.LogonExchangeMailbox

Re: RDO objects and C#

3. Loop through the entries in the RDOSession.AddressBook.GAL;
4. For each RDOAddressEntry with DisplayType property equal to DT\_MAILUSER (= 0)
5. Call RDOSession.GetSharedMailbox passing that RDOAddressEntry object (get back RDOExchangeMailboxStore)
6. Loop through all subfolder and messages in the mailbox store stating with the RDOStore.IPMSRootFolder collection

Will there be a problem in step 3 and 4 if there is a multiple exchange server box in the domain and various users are configured with different exchange boxes. Will I still be able to get all user's RDOAddressEntry object in RDOSession.AddressBook.GAL or will it give only those users which are on the same exchange box as of the Windows user under which the application is running?

Also I assume that RDOSession.Logon using the existing profile of using RDOSession.LogonExchangeMailbox are exclusive. I guess I don't have to create the profile if it's not there and can use the RDOSession.LogonExchangeMailbox method instead. Is that right?

Many Thanks for your help.  
Amit Kathane

"Dmitry Streblechenko" wrote:

Once again, see steps 1 and 2. You need to call LogonExchangeMailbox \*once\*

Re: RDO objects and C#

for the current Windows  
user who must have the  
right to access othe  
rmailboxes.

To access other mailboxess  
call  
RDOSession.GetSharedMailbox.  
Do \*not\*  
call  
LogonExchangeMailbox for  
the other mailboxes.

Dmitry Streblechenko  
(MVP)  
<http://www.dimastr.com/>  
OutlookSpy – Outlook,  
CDO  
and MAPI Developer Tool

"AK"

<AK@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

wrote in message

<news:23E06401-006F-4B39-8068-2E0000000000@xxxxxxxxxxxxxxxxxxxx>

Hi,

Thanks you  
very much  
for your  
reply.

I'm  
understanding  
bit of  
Redemption  
library now.

One issue is  
when I'm  
trying to do  
login using  
RDOSession.LogonExchangeMailbox

it's popping  
up the UI.

According  
to my  
requirement,  
I've to  
traverse

Re: RDO objects and C#

through all  
mailbox  
which  
I  
guess will  
require  
login into  
each  
mailbox and  
a pop up UI  
could be  
an  
issue?

This UI  
does not  
come when  
using  
RDOSession.Logon  
method, is  
there  
anything